CodonBiasGenerator.py User Guide

2015 UCSC iGEM Team

September 18, 2015

Table of Contents

List of Figures

General Overview

1.1 What is it for?

The CodonBiasGenerator program is a tool for providing an easy, parseable Codon Bias Table that can be read, analyzed, and manipulated according to the user needs. It works very much like the Codon Usage Frequency Table's found on GenScrpit and Kazusa. However, unlike the competitors the CodonBiasGenerator gives the user the option of any gene sequenced input while returning a better output in the form of a dictionary file. This file is then able to be used as a stardard text file capable of string and data manipulation.

1.2 Program Specifications

This program was written using the most recent version of the Python programming language, Python 3.4.3. It is designed to work with the sequenceAnalysis.py as an imported module using the FastAReader class as a sequence reader. The class of the program is CodonFreq, with three separate defining methods for easy access and variability, each corresponding to its own calculated/documented variable.

Class CodonFreq

This class was written by UCSC iGEM team members Raymond Bryan, Jairo Navarro, and Cristian Camacho. It was developed in the course BME 188A/B: Synthetic Biology Lab, and provides functionality congruently with sequenceAnalysis.py.

2.1 Attributes

- rnaCodonTable: a common dictionary used as a codon to AminoAcid(AA) key table
- dnaCodonTable: a revised rnaCodonTable dictionary using the translated RNA to DNA codons
- CodonCount: a listed dictionary organized as AA→ codon→ Value, where the value is the number of time the codon is accounted for within the Genome inputed.
- CodonFrequen: a listed dictionary organized as AA→ codon→ Value, where the value is the float usage Frequency of the codon in question calculated as codonCount/ total # AA is called
- CodonPerThou: a listed dictionary organized as AA→ codon→ Value, where the value is the codon Frequency per Thousand calculated as (codonCount/ Total # of codons)*1000

2.2 Methods

- readSeq(): This allows the each sequence read to be analysed of all codons and append their count to a master list dictionary by matching the codon at hand to the possible codons listed in the dictionary and finally consolidating the total number of times called into a single value as a dictionary output.
- TableMaker(): This allows the counted value gather by readSeq() to be accessed and used to determine the total amount of time the AA is called in the sequence by various possible codons into a separate dictionary . This new dictionary is then used to to calculate the Codon Usage Frequency, also stored as a dictionary output.
- PercentPerThou(): This allows the counted value gather by readSeq() to be accessed and used to determine the total amount codons called in the genome. This new value in conjuction with the individual codon counts, is then used to to calculate the Codon Percent per Thousand, also stored as a dictionary output.

Using the CodonBiasGenerator Program

The use of the CodonBiasGenerator can be done in either IDLE or CMD. Here are the following steps for making use of it in your system:

1. Download both files named sequenceAnalysis.py and CodonBiasGenerator.py from either the UCSC iGEM 2015 wiki, or the 2015 iGEM GitHub.
2. Important, save both files in the same directory you currently work with in python3.4
3. Open your favorite commandline and change to your working python directory
4. Type/ Run CodonBiasGenerator.py and enter your gene Fasta file
5. Witness python sorcery magic.

Below is an example program which illustrates the above steps and explains how to move further.